AF JW

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service for First-Class Mail Delivery, postage prepaid, addressed to: Mail Stop Appeal Brief – Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on 02 November 2005.

Lynne M. Milliot

Lynne M. Milliot

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Group Art Unit: 2176
Matthew FUCHS et al.	Examiner: Maikhanh NGUYEN
Application No.: 09/493,517	
Confirmation No.: 2359	
Filed: 28 January 2000	**CUSTOMER NO. 22470**
Title: **SYSTEM AND METHOD FOR SCHEMA EVOLUTION IN AN E-COMMERCE NETWORK**	

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

CORRECTED APPEAL BRIEF

Sir:

This Corrected Appeal Brief is filed in further support of Appellants' appeal from the Final Office Action, mailed 18 May 2005, in this case. A Notice of Appeal was filed on 18 August 2005. The initial Appeal Brief was filed on 18 October 2005.

An omission on page 1 and an inconsistency on page 13 are corrected with this submission. We believe that no fee is due. However, should it be determined that additional fees are required, the Commissioner is hereby authorized to charge those fees to Deposit Account No. 50-0869 (Attorney Docket No. JGR 1012-1).

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The real party in interest is JGR Acquisitions, Inc., the assignee of record.

II. RELATED APPEALS AND INTERFERENCES

There are no known appeals or interferences relating to this case.

III. STATUS OF CLAIMS

Claims 14-35 are pending in the application. All have been rejected and all of the rejections are subject to this appeal.

IV. STATUS OF AMENDMENTS

No amendments were filed subsequent to the Final Office Action (FOA) from which this appeal is taken.

V. SUMMARY OF CLAIMED SUBJECT MATTER

There are three independent claims, numbers 14, 25 and 31 which are addressed individually. Dependent claim 16 also is addressed. For this appeal, we group claims 15, 19 and 20-24 with claim 14. Claims 17 and 18 are grouped with claim 16. Claims 26-30 are grouped with claim 25. Finally, claims 32-35 are grouped with claim 31.

Claim 14 presents a method of extending a definition of a first tag used in a first electronic document, wherein the electronic document is encoded in a markup language, and the document is stored on a server in a computer network. The method includes three phases operating on data structures such as those illustrated in FIG. 2. First, defining the first tag in a first schema, wherein the definition of the first tag includes a plurality of elements from the markup language. Second, defining a second tag in a second schema, wherein the definition of a second tag includes the plurality of the elements from the markup language and an additional element from the markup language. And third, accessing the first schema and second schema in the first electronic document, wherein the first tag and the second tag are used to encode text within the first electronic document. *See, e.g.,* Application at pp. 14-15.

Claim 16 adds to claim 14 the limitation that the second tag is used in a place reserved for the first tag. *See*, Application at pp. 14-15.

Claim 25 describes a computer network system for processing a document instance of a markup language. This computer system comprises the following elements: a means for defining a first schema in the computer network system; a means for extending a definition of an element in the first schema by use of a second schema residing on the computer network system; and a means for importing the second schema into the document instance. Each of the elements of claim 25 is in means plus function form, the means including data structures in memory of a computer network system form processing a document instance. The structures corresponding to means for defining a first schema include an enhanced schema language, described on pp. 13 *et seq.* of the specification. *See,* FIG. 2. The structures corresponding to means for extending a definition of an element include the extends statement illustrated in the example on pp. 14-15. *See,* FIG. 2, ref 204. The structures corresponding to means for importing the second schema into the document instance include URNs, URIs and URLs directly or indirectly specified in an XML document, as illustrated on pp. 22-25. *See,* FIG. 2, ref 212, 214. Alternatively, the means include a processor responsive to data structures specifying definitions, extensions and imports.

Claim 31 describes a method operating in computer network system comprising a plurality of servers. The method applies to interpreting an XML document, the XML document residing on a first server from the plurality of servers. The method proceeds with accessing a first schema from a second server in the plurality of servers, wherein the first schema defines one or more elements used in the document instance. The method includes accessing a second schema from a third server in the plurality of servers, wherein the second schema extends at least one element from the one or more elements used in the document instance. *See, e.g.,* FIG. 2; Application at pp. 14-15.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Whether it was improper to reject claim 31 under 35 U.S.C. § 103(a) as being unpatentable over W3C, "XML Schema part 1: Structure," 06/05/1999 (W3C), as cited in Applicants' IDS filed on 12 November 2002?

Whether it was improper to reject claim 25 under 35 U.S.C. § 102(a) as anticipated by W3C?

Whether it was improper to reject claim 14 under 35 U.S.C. § 103(a) as being unpatentable over W3C?

Whether it was improper to reject claim 16 under 35 U.S.C. § 103(a) as being unpatentable over W3C?

VII. ARGUMENT

A. Preliminary Review of the Technology Disclosed and References

Much electronic commerce is conducted by Web Service modules that exchange XML documents. XML documents differ from prior technologies because they are parsable, field tagged documents. Well-formed XML documents typically are validated against a schema. Several schema languages have competed for adoption. Most competing schema languages offer new features not found in prior XML schema languages.

Appellants have disclosed technology useful for evolution of schemas, as industry faces the inevitable customization and versioning of XML documents. "Schema evolution" is in the title of the application. On page 27, safe extension and evolution of schema elements is described as an application of the disclosed technology.

The primary reference cited in the FOA is published by the industry standards organization W3C, on the web at **www.w3.org**. The "XML Schema" reference, which we will ususally refer to as "W3C", expressly disclaims addressing schema evolution, both at the end of § 3.1 and as an open issue in Appendix G, on the last page of the document. It says, "This draft does not deal with the requirement 'for addressing the evolution of schemata.'" *Id.*, § 3.1.

The following technology tutorial is intended to help the Board appreciate the field of technology and the well-established distinctions that were missed in the FOA.

1. The Disclosed Technology

The disclosed technology includes ways of extending an element of an original schema and thereby extending a document defined by the schema, without editing the original schema and without breaking applications that depend on the original schema.

It is illustrated as an extension of the SOX schema definition language promoted by Commerce One, the company to which this application originally was assigned.

Pages 14-15 of the application illustrate one embodiment of extending an element in a document, using an extended XML schema. In this illustration, an <Address> element of a PurchaseOrder document defined by the schema PurchaseOrder.sox is extended by creating a new element <Contact> which is defined in the supplemental schema ContactAddress.sox. Use of the supplemental schema allows a business to substitute the <Contact> element in place of the <Address> element in a PurchaseOrder document and still have a well-formed and valid document. That is, the supplemental schema allows omission of <Address> and substitution of <Contact> in its place, even if the PurchaseOrder.sox schema makes <Address> a mandatory element for every PurchaseOrder document. Because the illustration on pp. 14-15 is a concrete example of concepts that we will discuss throughout this brief, we reproduce it below:

> Suppose that <Address> is utilized by document instances of type *PurchaseOrder* and that a particular trading partner ACME wishes to make a simple extension to the <Address> element used in *PurchaseOrder.sox* 200. In particular, ACME wishes to extend the *PurchaseOrder.sox* 200 schema to allow the <Address> element to contain telephone numbers. As illustrated below, the present invention enables such an extension of the <Address> tag; the polymorphism feature allows the extended <Address> tag to be used in instance documents of type *PurchaseOrder,* while preserving the integrity of the *PurchaseOrder.sox* schema and the existing instance documents of that type.
>
> The <Address> tag may be extended by using the SOX schema language to create a small document type *ContactAddress,* whose corresponding schema *ContactAddress.sox* 204 extends the *CBL.sox* 2 I6 definition of <Address> to include a telephone number. The 20 extended tag, or element, is referred to as <Contact>, and this element is defined in *ContactAddress.sox,* which is given as follows:

```
<schema uri = "ContactAddress.sox">
    <namespace prefix = "CBL" uri = "CBL.sox"/>
    <elementtype name = "Contact">
        <extends prefix = "CBL" type = "Addressv>
            <append>
                <element type = "PhoneNumber" occurs = "*"/>
            </append>
        </extend>
    </elementtype>
```

</schema>

The new document type ContactAddress includes an identifier 206 for CBL.sox 216. A document instance 208 of type PurchaseOrder incorporates the new <Contact> tag by import statements which reference the schemas ContactAddress.sox 204 and CBL.sox 216 respectively. Note that the <Contact> tag may be used in any place in the document instance reserved for the original <Address> tag.

From this example, one sees that the supplemental schema overrides selected elements of the original schema, supporting modification of the original document definition (*e.g.*, for a PurchaseOrder) by referencing the supplemental schema, without any need to edit, update or modify the original schema.

2. The W3C Reference

The W3C reference introduces a schema language with capabilities lacking in DTD schemas for XML, "a superset of the capabilities found in XML 1.0 document type definitions (DTDs.)" *Id.*, Abstract. In the late 1990s, W3C's XML Schema language rivaled Commerce One's SOX schema language. In XML Schema § 2.3, the authors give examples of their new language features: "XML Schema: Structures not only reconstructs the DTD constraints of XML 1.0 using XML instance syntax, it also adds the ability to define new kinds of constraints. For example, although the author of an XML 1.0 DTD may declare an element type as containing character data, elements, or mixed content, there is no mechanism with which to constrain the contents of elements to only character data of a particular form, such as only integers in a specified range." The extensions do not address so-called evolution of schemas, even though the issue was recognized.

Important to this application, at the end of § 3.1, the authors explain, "**Issue (no-evolution)**: This draft does not deal with the requirement 'for addressing the evolution of schemata' (see [XML Schema Requirements])." (The boldfacing in this quote is from the original document.)

The FOA focuses on the import feature described in § 4.4. While § 2.4 contains an informal definition of "Schema Import", it is hyperlinked to § 4.4, which discloses how Schema Import actually works. (The reference is available at **http://www.w3.org/1999/05/06-xmlschema-1/**.) One imports an original schema into a subsequent schema to save retyping. Documents must meet the original schema requirements to be valid; the

original schema definitions are not changed when they are imported into the subsequent schema. From § 4.4:

Example

```
<schema name='http://coolbits.com/someschema.xsd'>

  <!-- Establish 'other' abbrev for use in schema -->

<import schemaAbbrev='other'
schemaName='http://coolbits.com/otherschema.xsd'/>

  <!-- can use 'other' anywhere in schema body -->

<elementType name='myNewElement'>
  <choice>
      <elementTypeRef schemaAbbrev='other' name='myElement'/>
      <elementTypeRef schemaAbbrev='other' name='otherElement'/>
  </choice>
</elementType>

</schema>
```

A valid instance would look like the following, delta any namespace prefix changes:

```
<myNewElement xmlns='http://coolbits.com/someschema.xsd'
              xmlns:pref1='http://coolbits.com/otherschema.xsd'>
    <pref1:myElement/>
    <pref1:otherElement/>
</myNewElement>
```

The default namespace establishes that `myNewElement` is an element name in the default namespace, and the two subordinate elements are from the namespace associated with the `pref1` prefix.

From this example, one sees that the text of "otherschema.xsd" has effectively been copied into the text of "someschema.xsd" by the schema import. However, elements referenced from "otherschema.xsd" must be explicitly qualified when they are used. In the two blocks of the example, "other" is used first as the schema abbreviation for explicitly referencing elements from the imported schema; "pref1" is used in the second, valid instance example.

Following an XML Schema import, parts of an original schema can be used to build larger constructs in the subsequent schema, but none of the original schema is modified when it is imported into the subsequent schema. Nothing from the imported schema, "otherschema.xsd" is changed in the examples and none of the validity constraints of otherschema.xsd are overridden by the import. Copying in the schema text allows elements such as myElement or otherElement to be used in newly defined

compound elements, such as myNewElement, without retyping the original definitions. This is why the authors repeatedly say that schema evolution is not addressed in the W3C reference.

3. The Usdin Reference

The Usdin reference is a primer for those uninitiated to XML. It predates the W3C reference by several months. On page 128, col. 1, it refers to "several proposals for alternate and richer ways to define/describe the contents of XML documents", which presumably references work under way at W3C that led to the W3C reference. It lists ongoing efforts, including the Common Business Library (CBL) developed by Veo Systems, which merged into Commerce One. *Id.*, at 130 col. 2.

The inherent "extensibility" of XML is mentioned on page 131. In this context, extensibility "allows entire industry and academic groups[] to define a markup that describes the data instead of using HTML markup for making the data look good in a particular brand of browser." In substance, the reference introduces XML as a new way to format data and to build industry-wide, shared data formats.

Usdin says nothing about schema evolution; it is addressed to a naive audience at a more general level.

We hope that the Board finds this technology tutorial helpful.

B. Rejection of Independent Claim 31 Under Section 103(a) as Unpatentable over W3C was Improper

We begin with claim 31 because it expressly claims extending an element of a document definition:

> *In a computer network system comprising a plurality of servers, a method of interpreting an XML document, the XML document residing on a first server from the plurality of servers, the method comprising:*
>
> *accessing a first schema from a second server in the plurality of servers, wherein the first schema defines one or more elements used in the document instance;*
>
> *accessing a second schema from a third server in the plurality of servers, wherein the second schema extends at least one element from the one or more elements used in the document instance.*

The claimed invention is not covered by W3C, as the reference expressly disavows addressing schema evolution.

Schema evolution is supported by the technology in claim 31. In that claim, the second (supplemental) schema extends at least one element defined in the first (original) schema. In this context, "extends" is understood from the specification, including pp. 14-15, to mean modifying, overriding or adding to an element found in the original schema, when the second schema is accessed to supplement the original schema. *See*, Application at p. 5.

The Examiner rejects claim 31 citing the Schema Import feature of W3C XML Schema. The analysis (FOA 14) is superficial:

> (ii) accessing a second schema *(e.g., XML schemas are themselves specified as XML documents ...schema definition documents are accessed during processing; Section 2.1)*,wherein the second schema modifies at least one element from the one or more elements used in the document instance *(e.g., extends the current schema with definitions and/or declarations from an external schema; Section 2.4 and 6.2).*

Tying the Examiner's language to the technology tutorial above, the subsequent schema uses definitions and/or declarations by importing from original schema. The second schema is the subsequent schema, because it is written to incorporate by reference the first (original) schema.

The Examiner's argument does not read on the *accessing a second schema* element of claim 31. The Examiner says the second (current) schema uses the first (external) schema, but that is not the same as *the second schema extend[ing] at least one element* defined in the first schema. There is a relationship between the importing and imported schemas of W3C § 2.4, but the second schema invoking the import does not extend an element defined in the first schema. The element definitions imported with the first schema are available to use, but unchanged and not extended by the import.

In one respect, the Examiner has things backward: he is looking at how the first schema becomes available for drafting the second schema. Claim 31 instead

addresses how the second schema overrides an element defined in the first schema. In another respect, the Examiner did not apply the reference to extending *at least one element* of the first schema that is used in the document instance, arguing instead how the second schema takes advantage of the first schema.

Focusing on § 2.4, the function of W3C's import is limited to saving on retyping. That may be why the W3C authors note that the reference does not address schema evolution.

For completeness, we mention §§ 2.1 and 6.2, neither of which have anything to do with schema evolution or claim 31. Section 2.1 defines terminology. Section 6.2 outlines responsibilities of schema-aware processors, with the authors' express caveat that it is not consistent with the rest of the XLM Schema specification. Neither of the sections section of W3C are remotely relevant to claim 31.

This is sufficient reason to find claim 31 allowable over W3C.

Next, the Examiner argues (FOA 7-8) that what W3C does not disclose is obvious from requiring a schema-aware XML processor to include an external processing system that has access to the document plus schema information. There is nothing inherent about putting the first and second schemas on different servers, because it would be more conducive to development for both schemas to be on the same server and more conducive to operations to have them on a single server controlled by the developer. The passage quoted in paragraph (b) (FOA 7) and reasoning in paragraph (d) (FOA 8) do not render obvious the claimed positioning of the first and second schema. So the quoted passage of § 6.2 proves nothing.

To the extent that paragraph (c) might be construed as a statement of the Examiner's own knowledge, which would be a stretch, in this first paper filed after the FOA, Applicants point out that the Examiner has failed to supply the required affidavit testifying to his own personal knowledge. MPEP ¶ 2144.03. For lack of supporting evidence, on appeal, paragraph (c) cannot be treated as anything more than argument.

For the two reasons given, claim 31 should be allowable.

B. Rejection of Independent Claim 25 Under Section 102(a) as Anticipated by W3C was Improper

Claim 25 is written in means-plus-function format:

A computer network system for processing a document instance of a markup language, the computer system comprising:

means for defining a first schema in the computer network system;

means for extending a definition of an element in the first schema by use of a second schema residing on the computer network system;

means for importing the second schema into the document instance.

The Examiner did not attempt to read W3C on any means described in the specification.

Curiously, the Examiner dropped the words "means for" from each of the elements (FOA 2-3). Doing so, the Examiner failed to make out a *prima facie* case in the terms required by case law and explained in MPEP § 2183.

For the sake of compact prosecution, Applicants point out that the example on pp. 14-15 of the application, set forth *supra*, at 4, depicts one structure *for extending a definition of an element in the first schema by use of a second schema.* An explicit "extends" statement overrides the original element definition in the first schema, for documents that reference the second schema. This is a means of supporting schema evolution. W3C does not have an equivalent override structure and expressly disavows support for schema evolution.

Rejection of independent claim 25 under section 102(a) should be reversed because the Examiner did not make out a *prima facie* case and there is no credible anticipation argument to be made from the W3C reference.

C. Rejection of Independent Claim 14 Under Section 103(a) as Unpatentable over W3C was Improper

In claim 14, a tag includes a plurality of elements and a tag is extended by the second schema, as we did for elements in claim 31. Claim 14 provides:

A method of extending a definition of a first tag used in a first electronic document, wherein the electronic document is encoded in a markup language, and the document is stored on a server in a computer network, the method comprising:

defining the first tag in a first schema, wherein the definition of the first tag includes a plurality of elements from the markup language;

defining a second tag in a second schema, wherein a definition of the second tag includes

the plurality of elements from the markup language; and

an additional element from the markup language;

accessing the first schema and second schema in the first electronic document, wherein the first tag and the second tag are used to encode text within the first electronic document.

These elements are not met by W3C.

The Examiner's argument (FOA 4) is that associating an Archetype Definition with an element anticipates this claim, referencing W3C § 3.4.9. To understand Archetype Definitions, we being with § 3.4.2 and the definition:

> [Definition:] Archetype definitions gather together all SCs pertinent to elements in instance documents, their attributes and their contents. They are called archetypes because there may be more than one element type which shares the same SCs (see Element Type Declaration), and which therefore can be constrained by a common archetype.

Section 3.4 uses the abbreviation "SC" for schema constraints and information set contributions that help define elements constrained by a common archetype. The information set contribution example of an archetype and SC given in § 3.4.2 involves the elements ShippingAddress and BillingAddress which share the sub-elements: Street, City and Zip. W3C explains, "By defining a single appropriate achetype and declaring two element types which reference it, that commonality can be precisely expressed." Archetypes are like macros, another way to save on typing.

For completeness, we reproduce the Examiner's argument (FOA 4) below. In parts (i) and (ii) of the argument reproduced below, the Examiner implies that a particular archetype is available in two different schemas. Then, in part (iii), the Examiner seems to confuse schemas with a *first electronic document* that uses *the first tag and the second tag ... to encode text within the first electronic document.* There is not sufficient explanation of the Examiner's position to make out a *prima facie* basis for an obviousness rejection, as one can see from the passage reproduced below.

(i) defining the first tag in a first schema, wherein the definition of the first tag includes a plurality of elements from the markup language *(e.g., an element type declaration associates a name with an Archetype Definition ... will appear in tags in instance document; section 3.4.9)*;

(ii) defining a second tag in a second schema, wherein a definition of the second tag includes the plurality of elements from the markup language *(e.g., an element type declaration associates a name with an Archetype Definition ... will appear in tags in instance document; section 3.4.9)* and additional element from the markup language (*e.g., expanding any entity references whose XML 1.0 declarations; section 6.2)*; and

(iii) accessing the first schema and second schema in the first electronic document *(e.g., XML schemas are themselves specified as XML documents ...schema definition documents are accessed during processing; Section 2.1)*

Because use of archetypes does not render claim 14 obvious, the rejection should be reversed.

D. Rejection of Dependent Claim 16 Under Section 103(a) as Unpatentable over W3C was Improper

Claim 16 depends from claim 15 and claim 14, in turn:

15. The method of claim 14, further comprising:

parsing the first electronic document, wherein the first electronic document is parsed by a parser for the markup language, the parser being stored on the server.

16. The method of claim 15, wherein the second tag is used in a location reserved for the first tag in the electronic document.

The ability of the second tag to override the first tag is not obvious from W3C.

The Examiner's rejection of claim 16 is most succinct, repeating the words of the claim and citing W3C § 3.4.1, without a word of explanation. Section 3.4.1 is entitled "Datatype Definitions." Citing the definition of simple datatypes in XML appears to be completely off point.

Recall the element <Contact> from the example on pp. 14-15 of the application. <Contact>, found in the supplemental schema, extends <Address> in the original schema. The application explains that a document that references the supplemental schema may substitute a <Contact> element in place of an <Address> element in a PurchaseOrder document, even though the base PurchaseOrder.sox schema makes the <Address> element mandatory. That is one sense in which a *second tag* may be *used in a location reserved for the first tag.* The Examiner's sense of this claim is elusive and not sufficiently well explained to make out a *prima facie* case of obviousness.

Therefore, rejection of claim 16 should be reversed.

CONCLUSION

In view of the foregoing, Applicant/Appellant asks that this honorable Board reverse the Examiner's rejections of the claims. In addition, it is submitted that all claims are now allowable, and a notice of intent to issue a patent is respectfully requested.

The Commissioner is hereby authorized to charge any fee determined to be due in connection with this communication, or credit any overpayment, to our Deposit Account No. 50-0869 (File No. JGR 1012-1).

Respectfully submitted,

Dated: 02 November 2005

Ernest J. Beffel, Jr., Reg. No. 43,489
Attorney for Patent Owner

HAYNES BEFFEL & WOLFELD LLP

P.O. Box 366
751 Kelly Street
Half Moon Bay, CA 94019
Telephone: 650.712.0340
Facsimile: 650.712.0263

CLAIMS APPENDIX

1. (Withdrawn) A computer network system for processing electronic documents encoded in a markup language, the computer network system comprising:

a communications channel;

a first server, the first server being in communication with the communications channel, wherein the first server stores a first schema, the first schema including a definition for a first element in the markup language, the definition of the first element further including a first sub-element in the markup language;

a second server, the second server being in communication with the communications channel, wherein the second server stores a second schema, the second schema including a definition for a second element in the markup language, the definition of the second element further including

the first sub-element

a second sub-element in the markup language.

2. (Withdrawn) The computer network system of claim 1, wherein the communications channel includes a local area network (LAN).

3. (Withdrawn) The computer network system of claim 2, wherein the local area network further includes an Ethernet LAN.

4. (Withdrawn) The computer network system of claim 1, wherein the communications channel includes a wide area network (WAN).

5. (Withdrawn) The computer network system of claim 1, wherein the communications channel includes the Internet.

6. (Withdrawn) The computer network system of claim 1, wherein the markup language comprises XML.

7. (Withdrawn) The computer network system of claim 6, further comprising:

a third server, the third server in communication with the communications channel, wherein the third server stores a first XML document instance, wherein the first document instance is interpreted by use of the first schema.

8. (Withdrawn) The computer network system of claim 7, wherein the first document instance includes the first element.

9. (Withdrawn) The computer network system of claim 7, wherein the first document instance includes the second element, such that the second element is used in a location reserved for the first element in the first document instance.

10. (Withdrawn) The computer network system of claim 9, further comprising:

a fourth server, the fourth server in communication with the communications channel, wherein the fourth server stores a second XML document instance, wherein the second document instance is interpreted by use of the second schema.

11. (Withdrawn) The computer network system of claim 10, wherein the second document instance includes the second element.

12. (Withdrawn) The computer network system of claim 11, wherein the second element is used in a location reserved for the first element in the second document instance.

13. (Withdrawn) The computer network system of claim 10, wherein the first document instance and the second document instance correspond to a document type, wherein the document type is at least one of a purchase order, a purchase order acknowledgement, an order status check, an availability check, a price check, an invoice, an invoice acknowledgement.

14. (Previously presented) A method of extending a definition of a first tag used in a first electronic document, wherein the electronic document is encoded in a markup language, and the document is stored on a server in a computer network, the

method comprising:

defining the first tag in a first schema, wherein the definition of the first tag includes a plurality of elements from the markup language;

defining a second tag in a second schema, wherein a definition of the second tag includes

the plurality of elements from the markup language; and

an additional element from the markup language;

accessing the first schema and second schema in the first electronic document, wherein the first tag and the second tag are used to encode text within the first electronic document.

15. (Original) The method of claim 14, further comprising:

parsing the first electronic document, wherein the first electronic document is parsed by a parser for the markup language, the parser being stored on the server.

16. (Original) The method of claim 15, wherein the second tag is used in a location reserved for the first tag in the electronic document.

17. (Original) The method of claim 16, wherein the markup language is XML.

18. (Original) The method of claim 17, wherein the first document corresponds to at least one of a purchase order, a purchase order acknowledgement, an order status check, an availability check, a price check, an invoice, an invoice acknowledgement.

19. (Original) The method of claim 14, wherein the first electronic document includes the first tag and the second tag.

20. (Original) The method of claim 14, further comprising:

accessing the second schema in a second electronic document, wherein the second tag is used to encode the second electronic document.

21. (Original) The method of claim 20, further comprising:

parsing the second document, wherein the second electronic document is parsed by a parser for the markup language, the parser being stored on the server.

22. (Original) The method of claim 21, wherein the markup language is XML.

23. (Original) The method of claim 22, wherein the second document corresponds to a commercial transaction.

24. (Original) The method of claim 23, wherein the commercial transaction is selected from the group consisting of a purchase order, a purchase order acknowledgement, an order status check, an availability check, a price check, an invoice, an invoice acknowledgement.

25. (Previously presented) A computer network system for processing a document instance of a markup language, the computer system comprising:

means for defining a first schema in the computer network system;

means for extending a definition of an element in the first schema by use of a second schema residing on the computer network system;

means for importing the second schema into the document instance.

26. (Original) The computer network system of claim 25, wherein the markup language is XML.

27. (Original) The computer network system of claim 25, wherein the definition of the first schema includes a definition of a tag.

28. (Original) The computer network system of claim 27, further comprising:

means for extending the definition of the tag by use of the second schema.

29. (Original) The computer network system of claim 28, wherein the document instance includes the tag.

30. (Original) The computer network system of claim 28, further comprising:

means for using an extension of the tag in the document instance, wherein the extension of the tag is used in a location reserved for the tag in the document instance.

31. (Previously presented) In a computer network system comprising a plurality of servers, a method of interpreting an XML document, the XML document residing on a first server from the plurality of servers, the method comprising:

accessing a first schema from a second server in the plurality of servers, wherein the first schema defines one or more elements used in the document instance;

accessing a second schema from a third server in the plurality of servers, wherein the second schema extends at least one element from the one or more elements used in the document instance.

32. (Original) The method of claim 31, wherein the computer network system is used to conduct a commercial transaction between two or more trading partners.

33. (Original) The method of claim 32, wherein the XML document corresponds to the commercial transaction.

34. (Original) The method of claim 33, wherein the commercial transaction is one of a purchase order, a purchase order acknowledgement, an order status check, an availability check, a price check, an invoice, an invoice acknowledgement.

35. (Original) The method of claim 31, further comprising:

parsing the XML document, wherein the document is parsed by an XML Processor residing on a fourth server from the plurality of servers.